



LAUREA
AMMATTIKORKEAKOULU

Uuden edellä

Yrityksen verkkosivusto ja sisällönhallinta

- Case TMI Herkkukonditoria-Leipä

Grönroos, Patrik

2013 Espoo, Leppävaara

Laurea-ammattikorkeakoulu
Laurea Leppävaara

Yrityksen verkkosivusto ja sisällönhallinta

Grönroos, Patrik
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
toukokuu, 2013

Grönroos, Patrik

Yrityksen verkkosivusto ja sisällönhallinta

Vuosi	2013	Sivumäärä	26
-------	------	-----------	----

Toiminnallisen opinnäytetyön tavoitteena oli kehittää verkkosivujärjestelmä yritykselle nimeltä TMI Herkkukonditoria-Leipä. Yritys halusi helposti päivitettävän ja toiminnallisesti nykypäivään sopivan internet-sivukokonaisuuden. Sivuston parempaa toimintaa varten luotiin sille räätälöity sisällönhallintajärjestelmä (eng. CMS, Content management system). Sisällönhallintajärjestelmällä tarkoitetaan internet-sivun teknistä puolta, jonka avulla ohjataan pääsivuston toimintaa. Teknisen puolen lisäksi oli tärkeää saada sivuston ulkoasusta houkutteleva käyttäen yrityksen omia tuotteita ulkoasun osana. Pää tarkoituksena opinnäytetyössä oli selvittää, kuinka rakennetaan oikeaoppinen sivukokonaisuus alusta alkaen itse käyttämättä valmiita verkosta löytyviä tuotteita.

Tavoitteena oli rakentaa tarkkaan valittujen ohjelmistokehityksen työkalujen avulla sivusto, jota voidaan helposti päivittää tarvittaessa. Valitut lähteet auttoivat saamaan syvemmän ymmärryksen valituista työkaluista sekä niiden käytön eduista. Näin ollen saavutettiin tietämys siitä, kuinka luodaan toimiva kokonaisuus kotikoneille ja mobiililaitteille.

Opinnäytetyön tekemisessä on auttanut tekijän aikaisempi työskentely yrityksessä. Yrityksen alkuperäinen sivusto ei tukenut mobiililaitteita ilman erillistä päivitystä. Lisäksi sivustolta puuttui päivitysmahdollisuus. Yrityksen asettama tavoite opinnäytetyölle oli sivuston parempi ulkoasu ja käytettävyys.

Sisällönhallintajärjestelmän toiminnassa haluttiin keskittyä käytön kannalta oleellisiin toimintoihin. Ominaisuuksissa päädyttiin tiedotusten ja erillisaukioloaikojen lisäämiseen, poistoon ja muokkaukseen. Sivurakenne jaettiin kahteen eri alustaan. Sisällönhallintajärjestelmän toiminta jaettiin omaksi osa-alueeksi, josta tuettiin julkisen internet-sivuston toimintaa. Lisäksi otettiin huomioon mahdolliset uhkat verkkosivuston ja sisällönhallintajärjestelmän tietoturvaratkaisuissa.

Kohdeyrityksellä on tavoitteena helpottaa toimintaansa lisäämällä enemmän tietotekniikkaan perustuvia ratkaisuja heidän jokapäiväiseen toimintaansa. Kohdeyritys oli työn loppuvaiheessa tyytyväinen luotuihin ratkaisuihin ja uskoo, että heidän asiakkaansa hyötyvät uuden sivuston ominaisuuksista. Lisäksi uusien sivujen avulla yritys sai entistä paremman markkinointikanavan sekä heidän tarvitseman tuen nykypäivän verkkosivustojen ylläpitotarpeisiin.

Grönroos, Patrik

Content management system for a company website

Year	2013	Pages	26
------	------	-------	----

The purpose of this thesis was to develop a functional and easy-to-use website with a CMS (Content management system) for a bakery called Herkkukonditoria-Leipä. Content management system refers to the technical side of the page, which is used to control the main site. In addition to the technical side of the website it was important that the design of the website used the company's own products as a part of the layout. The objective was to create a website through the author's own effort and discover how it is possible to build the content management system without any finished products found on the internet.

The object was to build a website with accurately selected software development utilities, which allow the website to be updated with ease if necessary. This way the company will get a fresh website that it needs. The selected sources contributed to a deeper understanding of the benefits of using these utilities in order to create a working website for modern computers and mobile devices.

The author was invited to plan a new website for the company on the basis of his previous work. The problem of the previous website was that it did not support mobile devices without a separate update. Neither did the site make easy editing possible. The plan for a new site was featuring everything needed to obtain an easy-to-use experience and also an impressive layout.

Only the essential features were kept on the content management system, such as in adding, editing and deleting of new posts and opening hours. The website was created in two parts. The first part was the content management system, which was a separate system and had the purpose of backing up the main website. The information security of the CMS was also improved as a result of not allowing the main site to be damaged by an attack.

The company's objective is to facilitate their work by adding more IT-based solutions to their everyday operations. The company was satisfied with the thesis project and believes that their customers will benefit from the new features of the site. In addition the new site gives to the company a better marketing channel, as well as the necessary support for today's website hosting needs.

Keywords website, CMS, HTML, CSS, MySQL, design, functionality

Sisälllys

1	Johdanto	6
1.1	Toimeksiantaja	6
1.2	Tavoitteet	6
1.3	Menetelmien käyttö.....	7
1.4	Edellytykset onnistumiseen	7
2	Internet-sivuston tietoperusta	8
2.1	Notepad++	8
2.2	HTML	8
2.3	PHP ja tietoturva	10
2.4	MySQL	11
2.5	JQuery	12
2.6	CSS.....	13
3	Suunnittelu.....	13
3.1	Tietoturva ja sen käyttömetodit	14
3.2	Ulkoasu.....	15
3.3	Internet-sivusto	16
3.4	Sisällönhallintajärjestelmä ja sen toiminta.....	17
4	Toteutus	18
4.1	MySQL	18
4.2	PHP	19
4.3	JQuery	20
4.4	Ulkoasu.....	20
4.5	Kuvankäsittely	20
4.6	HTML ja CSS	21
5	Yhteenveto.....	23
	Lähteet.....	24
	Liitteet	26

1 Johdanto

Opinnäytetyön tavoitteena on kehittää leipomo- ja konditoriatuotteita valmistavalle TMI Herkkukonditoria-Leipä -yritykselle helposti päivitettävä ja toiminnallisesti nykypäivään sopiva internet-sivukokonaisuus sisällönhallintajärjestelmän avulla.

Sisällönhallintajärjestelmällä tarkoitetaan internet-sivun teknistä puolta, jonka avulla ohjataan sivuston toimintaa. Teknisen puolen lisäksi on tärkeää saada sivujen ulkoasusta houkutteleva käyttäen erityisesti yrityksen omia tuotteita ulkoasun osana. Tarkoituksena on toteuttaa sivut ja lisätavoitteena selvittää kuinka sivut rakennetaan itse ilman valmiita internetistä löytyviä sisällönhallintajärjestelmiä.

1.1 Toimeksiantaja

Tilaaajayritys on vuonna 1987 perustettu leipomo- ja konditoriatuotteita valmistava TMI Herkkukonditoria-Leipä. Yritys toimii Lohjalla ja on yksi pisimpään seudulla toimineista elintarvikeyrityksistä. Yrityksellä on tavoitteena helpottaa toimintaansa lisäämällä enemmän tietotekniikkaan perustuvia ratkaisuja päivittäisten prosessiensa hoitamiseen. Näin yritys, sekä heidän asiakkaansa hyötyvät uudistuneen sivun ominaisuuksista ja yritys saa entistä paremman markkinointikanavan.

1.2 Tavoitteet

Työn tavoitteena on sivuston rakentaminen ilman valmiiden sisällönhallintajärjestelmien käyttämistä. Internet-sivuston tavoitteellinen toteutus määräytyy kahteen eri alustaan; itse internet-sivustoon ja sisällönhallintajärjestelmään. Toimeksiantajayrityksellä on lisätavoitteena lisätä tuki mobiililaitteille. Näin mahdollistetaan useamman asiakkaan pääsy sivuille, sillä edellinen versio ei mahdollistanut helppoa käyttöä kyseisillä laitteilla. Muokkausominaisuudet tahdottiin rajata vain välttämättömiin toimintoihin eli niihin, joita varsinaisesti käytettäisiin.

Työssä hyödynnettiin niin Laurea-ammattikorkeakoulussa opittuja asioita kuin jo valmiita tietoteknisen osaamisen taitoja. Lisäksi työn vaiheissa halutaan paikantaa mahdollisia ongelmatilanteita jo ennen niihin törmäämistä. Tarkoituksena on parantaa ongelmanratkaisutaitoja sekä oppia entistä enemmän kuinka toimivat sivut rakennetaan oikeaoppisesti alusta alkaen itse, ilman turvautumista valmiiksi olemassa oleviin tuotteisiin, joita on tarjolla lukuisia.

1.3 Menetelmien käyttö

Työssä käytetty tietoperusta tukee vahvasti valittujen ohjelmistokehitysmenetelmien käyttöä. Näiden menetelmien avulla saadaan työn kulku ja oppimisprosessi mahdollisimman tehokkaaksi. Valitut ohjelmistokehityksen metodit tukevat sivuston ja sisällönhallinnan oikeaoppisessa suunnittelussa.

Tavoitteena oli rakentaa tarkkaan valittujen ohjelmistokehityksen työkalujen avulla sivusto, jota voidaan helposti päivittää tarpeen tullen. Tekijän valitsevat lähteet auttoivat saamaan syvemmän ymmärryksen valituista työkaluista sekä niiden käytön eduista. Näin ollen saavutettiin tietämys siitä, kuinka luodaan toimiva kokonaisuus kotikoneille ja mobiililaitteille. Työn muoto suunnattiin toimintakeskeiseksi, sillä valmis verkkosivujärjestelmä toisi yritykselle sen tarvitseman tuen nykypäivän verkkosivustojen ylläpitotarpeisiin.

1.4 Edellytykset onnistumiseen

Toimeksiantoa helpotti aiempi työkokemuksen kautta kertynyt osaaminen yrityksessä ja antoi mahdollisuuksia kehittää yrityksen toimintaa. Vuorovaikutteisuuden avulla kuitenkin mahdollistettiin se, että järjestelmästä kehittyi yrityksen kannalta sille parhain räätälöity kokonaisuus.

2 Internet-sivuston tietoperusta

Opinnäytetyön onnistumisen kannalta käytettiin lukuisia eri ohjelmia ja merkintäkieliä. Käytettyjen tapojen yhtenäinen tuki mahdollisti sivun teknisen sekä ulkoasullisen toiminnan palvelimella. Seuraavassa osiossa syvennytään näiden ohjelmistojen toimintaan sekä niiden ominaisuuksiin parhaimman lopputuloksen aikaansaamiseksi.

2.1 Notepad++

Notepad++ on ilmainen avoimeen lähdekoodiin perustuva koodieditori, eli muokkausohjelma. Se tukee useita eri merkistökieliä ja on suunniteltu pääsääntöisesti Microsoft Windowsin käyttöön. Notepad++ on kirjoitettu C++ -ohjelmointikielellä ja se käyttää Win32 API sekä STL toimintoja, mikä takaa tehokkaamman toimintanopeuden ja ohjelmakoon. (Notepad++. 2011)

2.2 HTML

HTML on merkistökieli, jolla luodaan verkkosivuja. HTML, eli hyper text markup language perustuu tagien, eli merkistöjen tuottamaan toimintarakenteeseen. Kuten W3Schools (2013) kertoo, tagit määrittelevät sisällön toimintarakenteita. Toimintarakenteiden osat määrittyvät `<>` -merkkien kautta. Ne toimivat aina pareittain aloittaen ja lopettaen määritetyn rakenteen. Alkurakenteen hakasuljepari, `<>` aloitetaan ilman lisämerkintöjä. Loppurakenne, `</>` lopettaa määrätyn toiminnan, joka merkitään vinoviivalla `/`. Näin tiedetään milloin tietty määräys alkaa ja loppuu. Merkistön `<>` sisällä oleva määritys, esim. `<HTML>` aloittaa merkityn toiminnan ja `</HTML>` lopettaa sen. (W3Schools 2013.)

Kun puhutaan HTML-tageista ja HTML-elementeistä tarkoitetaan niillä usein samaa asiaa. Tarkasti painottaen sanojen merkitystä, HTML-elementillä tarkoitetaan kaikkea aloitusmerkistön ja lopetusmerkistön välillä olevia tagi-merkintöjä. Esimerkkinä `<p> Tämä määräys sisältyy tähän </p>`. Näin kaikki tagien sisällä olevat määritykset ovat toiminnassa. (W3Schools 2013.)

W3Schoolsin (2013) dokumentaatiossa painotetaan internet-selainten tärkeyttä HTML-dokumenttien kanssa. Niiden toiminnan edellytys on lukea HTML-merkistöä ja näin luoda luetusta merkistöstä näkyvä sivusto. Selaimet kuten Google Chrome, Mozilla Firefox, Internet Explorer ja Safari lukevat HTML-dokumentteja ja näyttävät ne internet-sivustoina.

HTML:n kanssa on mahdollista luoda internet-sivusto vähällä vaivalla. Silti sen avulla on mahdollisuus tuottaa erityisen joustavaa ja suunnittelijakeskeistä työtä. Lisäksi se pitää

suunnittelijoiden mielenkiinnon yllä ja edesauttaa heitä kokeilemaan erilaisia vaihtoehtoja. (Grannel, Sumner & Synodinos 2012, 6)

Kehittyneempi ja ajanmukaisempi versio HTML -kielestä on HTML5, joka on päivitetty versio HTML-merkistöstä. HTML-merkistö on ollut ajankohtainen jo vuodesta 1997. Suurin osa nykypäivän internet-selaimista tukee kyseistä merkistöä. HTML5:n muutokset ovat merkitysoipillisia eli osa merkistöstä on muutettu tai päivitetty. Näitä muutoksia ovat muunmuassa <article> -elementti ja sen lisäksi kehittyneemmän grafiikan näyttämiseen suunniteltu <canvas>. <video> lisää tuen videoiden toistamiseen HTML-dokumentissa. (Grannel, Sumner & Synodinos 2012, 8)

HTML5 on suunniteltu yksinkertaisemmaksi, joustavammaksi ja kehittäjäystävällisemmäksi. Lisäksi sen yhteensopivuus muissa laitteissa on kehittyneempi kuin edeltäjiensä HTML4 tai XHTML:n. Uusia toimintoja ovat muun muassa animointi, offline-tila sekä äänien lisäys. Lisätoimintoja edellisten lisäksi on myös graafisen käytön parannukset, kuten typografia eli kirjaisimien käyttö. HTML5 tuo internet-sivustoille uuden alun web-standardien parissa käyttökokemusten parantuessa. (Grannel, Sumner & Synodinos 2012, 6)

Näillä edellä mainituilla asioilla voidaan merkittävästi haastaa aikaisemmin hallinnut Flash-ohjelmointikieli. Flash-ohjelmointi on pitkään ollut johtava kehitysympäristö käyttäjän kannalta syventävien kokemusten ja interaktiivisesti upeiden sivujen toteuttamisessa. (Adobe Flash 2013.)

Hogan (2010, 15) toteaa, että Internet-sivustojen suunnittelijat ovat aina halunneet luoda entistä parempia käyttökokemuksia verkossa. HTML5 tarjoaa nykyisin myös hämmästyttäviä toimintoja, jotka joissain tapauksissa eliminovat tarpeen käyttää kolmannen osapuolen tarjoamia työkaluja, kuten Flashia. Vaikkakin Flashin käyttö on suhteellisen helppoa, ei se valitettavasti toimi suoraan esimerkiksi tietokone- ja mobiililaittevalmistaja Applen tuotteissa. Tämä jättää suhteellisen ison markkinaraon mobiilimarkkinoilla. Muiden tekniikoiden hyödyntäminen, HTML5:n tapaan, edesauttaa mobiililaitteissa toimivien sivustojen määrää.

Suurin syy HTML5:n suosioon nykypäivänä on myös sen toimivuus suurimmassa osassa internet-selaimista. Jopa Microsoftin kehittämälle Internet Explorer 6:lle voidaan kehittää sivuja HTML5:n avulla. Viidennen version kehittymisen myötä validointisivustot eli sivuston oikeaoppisen toiminnan tarkastavat sivut, kuten W3C tukevat ja validoivat suunnittelijan kehittämiä sivustoja. (Grannel, Sumner & Synodinos 2012, 6).

Grannel, Sumner ja Synodinos (2012, 17) kuitenkin täsmentävät, että vaikka HTML5 ei ole vielä saavuttanut täyttä tukea validointisivustojen ja selainten parissa. Se tulee kuitenkin

jatkamaan kehitystään ja pitää suosituimmat internet-selaimet kuten Googlen Chromen, Mozilla Firefoxin ja Safarin mukanaan. HTML5 merkistön etenevä kehitys tulee eittämättä aikanaan aiheuttamaan standardien muuttumista ja sitä kautta vanhojen standardien käyttämien sivustojen toimimattomuutta.

2.3 PHP ja tietoturva

PHP eli hypertext preprocessor on palvelin pohjainen komentosarjakieli, joka on suunniteltu luomaan websivustoista dynaamisempia. PHP-komentosarjakieltä voidaan sisällyttää HTML-merkistön avulla tehdyille sivulle ja suorittaa se, kun sivulla käydään. Kun käyttäjä käy sivuilla, ei PHP-merkistöä voi nähdä, vaan se toimii huomaamattomasti taustalla. Tätä kutsutaan tarkemmin palvelimella puolella toimivaksi toiminnoksi. (eng. Server-side) (Thomson & Welling 2009, 3)

PHP on avoimen lähdekoodin ohjelmisto, joka tarkoittaa sen olevan avoinna kaikille sitä tarvitseville. Tämän hetken käytetyin versio on PHP5, joka uudelleenkehitettiin uusien metodien avulla ja toi täten merkittäviä parannuksia merkintäkieleen. (Thomson & Welling 2009, 3)

Thomson ja Welling (2009, 5) kertovat PHP-merkintäkielen vahvuuksien olevan muunmuassa sen suorituskyvyssä. PHP on erittäin nopea prosessoimaan tietoa ja heikkoa palvelinta käytettäessä se voi prosessoida miljoonia kävijöitä päivässä. PHP tukee myös monia eri relaatiotietokantajärjestelmiä, muunmuassa MySQL-relaatiotietokantajärjestelmää.

Erityisesti Thomson ja Welling (2009, 6) pitävät sen tarjoamasta tuesta useille eri järjestelmillä. PHP toimii useissa eri Linux-distribuutioissa eli jakeluversioilla sekä Windows-käyttöjärjestelmillä. PHP on myös aloittelijalle erittäin helppo opetella.

PHP5 on edellisiä osiaan kehittyneempi, jonka takia se prosessoi tietoa tehokkaammin. Lisäksi mainitsemisen arvoisia parannuksia ovat syötteiden suodatus parantuneen tietoturvan avulla sekä javascriptin parempi tulkinta. (Thomson & Welling 2009, 6).

PHP:n huono tietoturva johtuu Snyderin, Myerin ja Southwellin (2010, 35) mukaan sen huonosta suunnittelusta. Lähtökohtaisesti ongelma on, ettei se tee mitään muuta kuin mitä sen käsketään tehdä. Tästä syystä vahinkoja voi sattua, jotka myöhemmin kostaavat heikkona tietoturvana. Siksi täytyykin ottaa selvää asioista etukäteen vahinkojen ehkäisemiseksi.

Verensin (2010, 56) mukaan PHP käyttää edelleen paljon vanhanaikaisia metodeja, jotka jakavat syötetyn tiedon globaalina tietona. Koodi eli ohjelmointitapa, jolla määrätään tietokoneohjelman toiminta (Bolton 2013) ajetaan usein avoimessa ympäristössä eikä funktioiden tai luokkien sisällä. Tämä tarkoittaa koodin olevan avoinna kaikille ja pahimmassa tapauksessa se jaetaan suoraan verkon kautta ilman suojausta. Tavan käyttäminen oli ennen suosittua, sillä se oli helpoin tapa tuottaa toimivaa koodia. Nykyään kehittäjät tekevät kaikkensa pystyäkseen ehkäisemään tiedon syötön avonaisessa ympäristössä. Bramptonin (2008, 112) mukaan hyödyllisin tapa tarkastaa epäkohtia esimerkiksi tietokannoissa on käyttää PHP5:n toimintoa, joka seuraa ja raportoi poikkeuksista. Kun virhe tapahtuu, lähettää se myös poikkeuksen, joka kirjoitetaan joko tietokantaan tai käytettävälle sivulle.

SQL -injektio on Cabralin ja Murphyn (2009, 657) mukaan yksi eniten huolenaihetta aiheuttavista asioista PHP:n ja MySQL:n kanssa toimivassa kokoonpanossa. Yksinkertaisesti injektio tapahtuu kun hyökkääjä lähettää turvaamattomaan lomakkeeseen sen toimintaan kuulumattoman kyselyn. Kysely tarkoittaa mielivaltaisten sql-lausekkeiden ajamista tietokannassa. (Freeservers 2013). Kysely voi pahimmassa tapauksessa poistaa koko tietokannan. Cabral ja Murphy (2009, 657) kuitenkin muistuttavat, että hyökkääjä voi vain tehdä asioita, joita tietokannan käyttäjälle on sallittu. Siksi normaalikäyttäjälle suositellaan rajoitettuja oikeuksia.

SQL -injektion suojaamiseksi on kehitetty keinoja, jotka vähentävät riskiä joutua väärinkäytön uhriksi. Useinmiten käytetty tapa ehkäistä väärinkäyttö on lisätä PHP-tietolomakkeeseen `mysqli_real_escape_string()` funktiot, jolloin tietokanta ei hyväksy kyselyitä ulkopuolisilta lähteiltä. (Snyder, Myer, Southwell 2010, 37)

Lisäksi Snyder, Myer ja Southwell (2010, 40-41) kertovat, että MySQLi -laajennus tarjoaa laajan skaalan erilaisia funktioita, joiden avulla erilaisten kyselyiden suunnittelu on turvallisempaa. Laajennus tarjoaa suunnittelijalle juuri sen tuen, jota tarvitaan tuottamaan omat turvalliset `safe()` -funktiot. Laajennuksen käyttämistä suositellaan erityisesti, sillä sen avulla yhdistetään sivuston toiminta tietokantaan.

2.4 MySQL

MySQL on relaatiotietokantajärjestelmä, johon voidaan tallentaa suuria määriä organisoitua tietoa helpon käytön takaamiseksi. PHP on MySQL:n kanssa käytetty yleisin merkistökieli, joka toimii yhteistyössä relaatiotietokannan kautta. Sen avulla voidaan hakea tai tallentaa haluttu tieto tietokantaan ja sitä kautta viedä se internet-sivustolle. (Yank 2012, 18)

MySQL relaatiotietokannasta voidaan edellisten ominaisuuksien lisäksi etsiä tallennettua tietoa. MySQL toimii palvelimelle asennettuna ohjelmistona. Ohjelmiston avulla tietoa

voidaan jakaa monille eri käyttäjillä helpolla ja nopealla tavalla. Tietokanta käyttää SQL, eli Structured Query Languagea, joka on alan standardi. (Thomson & Welling 2009, 3)

MySQL on ollut käytettävissä vuodesta 1996 lähtien, mutta sen kehityshistoria ulottuu aina vuoteen 1976 saakka. Se on maailman suosituin avoimen lähdekoodin tietokanta ja on voittanut jopa arvostetun Linux Journal Readers' palkinnon useita kertoja. (Thomson & Welling 2009, 3)

Suorituskyvystä Thomson ja Welling (2009, 8) kertoo, että MySQL tietokannan vahvuuksia ovat erityisesti sen toiminnan nopeus. Se on kiistämättä nopea ja on todistanut sen ollessaan vertailussa maksullisen kilpailijan Oracle -tietokantajärjestelmän kanssa. MySQL tukee myös useita eri käyttöjärjestelmiä, kuten Linuxia ja Microsoft Windowsia.

2.5 JQuery

JavaScript-kirjastoon perustuva JQuery on suosittu avoimen lähdekoodin ohjelmointiympäristö (Oracle 2013). Alkuperäinen Javascript on verkkosovelluksille tarkoitettu merkistökieli, joka käyttää erilaisia kirjastoja eli tiedostoja sen toiminnan osana. Sen toimintaperiaate ja merkistöjen käyttö on kuitenkin hankalaa, ja vaatii usein laajaa tuntemusta. (Jscripters, 2010)

JQuery on ollut helpotus niin ammattilaisille kuin aloittaville kehittäjille. Sen toiminta perustuu edellä mainittuun Javascript-kirjastoon, joka yksinkertaistaa tavan ohjelmoida kyseistä merkistökieltä. Jscriptersin (2010) mukaan JQuery tuo mukanaan useita parannuksia, joiden avulla voidaan kirjoittaa helposti aikaisemmin vaikeina pidettyjä ohjelmointirakenteita.

Jscripters (2010) kertoo JQueryn hyödyistä muun muassa jo aikaisemmin mainitulla helppokäyttöisyydellä. Helppokäyttöisyyden lisäksi ohjelmointikoodia ei tarvitse enää tuottaa yhtä paljon Javascriptiin verrattuna. Vaikkakin JQuery on melko uusi menetelmä, on sillä vahva tuki kehittäjien keskuudessa. Verkosta löytyy useita satoja valmiiksi kirjoitettuja lisä-osia ja opetussivustoja, jotka tukevat kehitysprosessia entisestään.

Sivuston toimintaan ja prosessien hyötykäyttöön katsoen Javascripters (2010) mainitsee kehitysedun varsinkin AJAX-kehitysalustassa. Ajax mahdollistaa sivistyneemmän käyttöympäristön, jonka avulla toimintoja voidaan suorittaa ilman verkkosivun täyttämistä uudelleenlataamista.

JQuery tukee myös työskentelyä PHP:n ja MySQL-relaatiotietokantojen kanssa. Ajax-toiminta tukee funktioita, joiden avulla voidaan lähettää tietoa HTTP, GET ja POST-toimintoja tukien. (Oracle 2013.)

Hyödyistä huolimatta, JQueryyllä on myös haittapuolia. Jscriptersin (2010) mukaan suuresta käyttökirjastosta huolimatta, sen toiminnot on vielä rajattu. Riippuen siitä, kuinka paljon muokattavuutta halutaan kehitettävälle sivustolle, johtaa se usein normaalin Java-scriptin käyttöön JQueryn kanssa. Lisäksi toimiakseen JQuery vaatii sivun sisältävän oman tiedostonsa, joka tuottaa pientä suorituskyvyn menettämisen palvelinpuolella. JQuery on kaikesta huolimatta kuitenkin suositeltava valinta verkkosivualustojen kehitykseen.

2.6 CSS

CSS (Cascading Style Sheets) on W3C:n standardi internet-sivustojen visualisoimiseen eli sivuston näyttävämmän ulkonäön saamiseksi. Sen käsite on yksinkertainen, mutta mullistava. Sen avulla erotetaan muotoilu omaksi tiedostokseen, jonka internet-sivusto hakee automaattisesti saadakseen määritetyt tyyliasetukset. CSS:ä voidaan sisällyttää suoraan HTML-tiedostoon. Erillisen tiedoston luominen säästää rahaa sekä asiakkaalta että suunnittelijalta ja helpottaa sivustolle tehtävien muutosten toteuttamista. (Grannel, Sumner & Synodinos 2012, 9)

Uudelleen kehitetty CSS3 on rikkaampi kehitysympäristöltään kuin aikasempi CSS-standardi. Sen avulla voidaan rakentaa entistä kevyempiä sivurakenteita, joita miltei jokainen nykypäivän internet-selain pystyy prosessoimaan. Näistä voidaan luetella muun muassa Google Chrome, Mozilla Firefox, Safari, Opera ja jopa Internet Explorerin versiot 8 ja 9. CSS3:n käyttäminen verkkosivuprojekteissa tuo entistä rikkaamman tavan luoda upeita sivuja ja parempia käyttöliittymiä nopeammin. (Css3explained 2012)

CSS3:n avulla voidaan luoda sivustolle näyttäviä ominaisuuksia. Lightbox-ominaisuus on yksi niistä. Lightboxin avulla saadaan luotua kuvagallerioita, joita klikkaamalla saadaan kuvan esitystapaa muutettua animoinnin seurauksena. Animoinnin viiveen lisäksi kuvan kokoa, ympyröivää värimaailmaa sekä läpinäkyvyyttä voidaan muokata. (Lou 2011).

3 Suunnittelu

Suunnittelun alkuvaiheessa pidettiin toimeksiantoyrityksen toimesta suunnittelukokouksia. Kokouksien avulla käytiin läpi nykyisen internet-sivuston vahvuuksia ja heikkouksia. Näiden summasta aloitettiin jatkokehityskeskustelut, joista saatiin parhaimmat parannusehdotukset esille. Kehityskeskusteluissa tuli esille muun muassa mobiililaitetuen tarve. Edellisen sivun

Flash-ohjelmoinnin takia sivusto ei ollut valmis toimimaan mobiililaitteissa ilman erillistä päivitystä älypuhelimille tai taulutietokoneille. Tämän seurauksesta päädyttiin muuttamaan koko sivuston ohjelmointirakennetta.

Vilches (2010) kirjoittaa Applen entisen toimitusjohtaja Steve Jobsin kirjoittamaan avoimeen kirjeeseen perustuen Applen olleen alusta asti Flashin käyttöä vastaan omista mobiililaitteissaan. Huonoiksi puoliksi lueteltiin muun muassa ohjelmoinnin sulkeutuneisuus, luotettavuus- ja tietoturvaongelmat sekä suorituskyvyn puute että akunkeston väheneminen. Tästä asiat kehittyivät vielä huonompaan suuntaan. Kyseinen yritys päätti lopulta kieltää kehittäjiä käyttämästä kolmannen osapuolen sovelluskehitystyökaluja laitteissaan. Applen virallinen kanta onkin, että se suosittelee menneisyyden sijaan Flashin luoja Adoben keskittyvän tulevaisuudessa HTML5 työkalujen kehittämiseen parempien käyttäjäkokemusten aikaansaamiseksi.

Kohdeyrityksen toiveisiin kuului myös, että edellisten sivujen ulkoasun värimaailma haluttiin säilyttää. Teknisessä toteutuksessa päädyttiin useiden lähteiden avulla HTML5-merkistön käyttämiseen pääalustana. Lisäksi paremman ulkoasun aikaansaamiseksi suunniteltiin CSS3:n käyttämistä.

3.1 Tietoturva ja sen käyttömetodit

Opinnäytetyön tietoturvassa otetaan huomioon asioita, joita tullaan tarvitsemaan toimivan kokonaisuuden aikaansaamiseksi. Metodeihin kuuluu muun muassa, lähetysten turvaaminen (PHP-tietoturva) ja käyttäjän tietojen salaaminen.

Useimmat ovat tietoisia internet-sivustoihin kohdistuvista uhkista. Pahin uhka on hyökkäys sivustolle, mutta myös pienemmät uhkat täytyy huomioida. Tehokkain tapa parantaa internet-sivuston tietoturvaa on suunnitella se perusteellisesti alusta lähtien ottamalla asioista selvää suunnittelun edetessä. Kehitysalusta saa näin sen ansaitseman parantuneen tietoturvan. Edellämainittu tapa edesauttaa myös tulevien tietoturvaratkaisujen lisäämistä valmiiseen kokoonpanoon. Toinen asia on virheilmoitusten huomioon ottaminen. Virheilmoitusten avulla voidaan ehkäistä sekä tulevia että tapahtuneita asioita, jolloin virheet voidaan korjata hyvissä ajoin. (M. Brampton 2008, 26)

Opinnäytetyössä käytettävää MD5 salausta PHP.net (2013) kuvaa tarkistetiedostona, joka muuttaa syötetyn tiedon 32-kirjaimiseksi hexadesimaalimeroksi käyttäen RSA Data Security -yrityksen kehittämää algoritmia tukena. Verens (2010, 37) tarkoittaa MD5 algoritmin olevan nopea tapa salasanojen suojaamiseksi. Yleisin tapa on muuttaa syötetty salasana PHP-lomakkeen avulla tietokantaan, jolloin se salataan PHP:n avulla 32-kirjaimiseksi

merkkijonoksi. Tapa on suosittu, sillä edes tietokannan pääkäyttäjä ei näe alkuperäisiä salasanoja. MD5 on kuitenkin tietoturvan kannalta heikko valinta, sillä jos hyökkääjä pääsee käsiksi tietokantaan, voi hän helposti saada selville kaikkien käyttäjien salasanat.

3.2 Ulkoasu

Oikeaoppisen ulkoasun suunnitteleminen lähtee ideointityöstä, joka on parasta tehdä ensin paperille. Useiden eri ideoiden vertaaminen paperilla ei ole vain nopeampaa vaan myös tehokkaampaa, kuin esimerkiksi tyhjästä aloittaminen graafisten ohjelmien parissa. Ideoiden vertaaminen useisiin suunnitelmiin antaa edun yhdistellä toimivat elementit ja saada loppuratkaisusta entistä toimivampi ennen siirtymistä lopullisen ulkoasun suunnitteluun koneella. (Grannel, Sumner ja Synodinos 2012, 26)

Grannel, Sumner ja Synodinos (2012, 26) tähdentää sisällön löytymisen tärkeyttä internet-sivustoilla. Navigoinnin ei kuuluisi missään tapauksessa olla vaikeasti löydettävissä vaan heti käyttäjän ulottuvissa. Yrityksen brändilogon olisi myös hyvä olla selkeästi esillä, jotta sivun sisältöä ei sekotettaisi. Etusivun kuuluu olla selkeä ja yksiselitteinen sekä siitä pitää selvittää sivuston pääkohtainen ajatus lyhyesti kerrottuna.

Grannelin, Sumnerin ja Synodinuksen (2012, 26) mukaan ulkoasun kuuluu olla eheä läpi koko sivuston, sillä vaihtuva ulkoasumaailma rikkoo käyttökokemusta. He kuitenkin tarkentavat, että vaihtuva ulkoasu toimii joskus, jos haetaan lehdentyyppistä ulkoasuratkaisua.

Yleisen käytettävyyden nimissä Grannel, Sumner ja Synodinos (2012, 26) ovat listanneet tapoja tarkastaa sivuston toimivuutta ottamalla huomioon seuraavia asioita; Onko sivustolla helppo navigoida? Onko sisällön etsiminen käyttäjälle helppoa? Ovatko latausajat tarpeeksi pienet? Onko sivu rakennettu sopimaan kohdeyleisön tarpeisiin? Lisäksi onko sivusto rakennettu totuttujen metodien mukaan? Jos nämä seikat täsmäävät on sivuston suunnittelu oikeilla jäljillä.

Sivuston suunnittelussa täytyy ottaa huomioon myös monia eri muotoiluseikkoja. On tiettyjä sääntöjä, kun valitaan taustakuvaa tai taustaväriä internet-sivustolle. Taustan ei kuuluisi häiritä koskaan sisältöä. Jos käytetään taustakuvia, tulisi ne pitää yksinkertaisina ja tiettyä väriä käytettäessä olisi suotavaa varmistaa toimivuus sivuston muun värimaailman kanssa. Tekstin alla ei saisi koskaan käyttää monitahoisia taustakuvia, sillä ne sekoittuvat tekstiin. Pehmeä gradient eli liukuväritausta toimii useimmissa tapauksissa erinomaisen hyvin. Kuvien koot eivät saa myöskään olla liian isoja, sillä sivuston latausaika kuuluu pitää pienenä. (Grannel, Sumner & Synodinos 2012, 42.)

3.3 Internet-sivusto

Kohdeyritys antoi vapaat kädet ohjelmointiympäristön valinnassa. Tämän ansiosta päädyttiin etsimään modernia ja helppokäyttöistä tapaa luoda mahdollisimman toimiva nykymaailmaan sopiva ratkaisu.

Ohjelmointiympäristössä päädyttiin käyttämään neljää yhdistävää nykyajan tapaa luoda dynaamisesti toimiva internet-sivusto (HTML, PHP, MySQL, CSS). HTML-kielellä suunniteltiin sivuston rakenne. Toteutustapa sivuston rakenteelle tapahtui useiden divien eli sisältöä merkitsevien tagien avulla. HTML5 valittiin optimoidun käytön aikaansaamiseksi nykypäivän älypuhelimille ja taulutietokoneille.

Työssä päädyttiin PHP:n ja MySQL:n yhteistyöhön niiden erinomaisen työhön soveltuvuuden ansiosta. Suurinosa PHP-koodista pystytään kirjoittamaan helposti käyttöjärjestelmien sekä palvelimien välillä. Lisäksi suurin osa käyttöjärjestelmistä, kuten Linux- ja Microsoft Windows versiot tukevat PHP:n ja MySQL:n käyttöä. (Thomson & Welling 2008, 3). Näin ollen se helpottaa tekemään työtä useilla eri kehitysalustoilla.

Kuten edellä mainittiin HTML-merkistön lisäksi sen kehittyneempää HTML5-merkistökieltä tullaan käyttämään. Lisäksi sivuston muotoilua tullaan parantamaan käyttämällä CSS3-merkistöä. Hoganin (2010, 14) mukaan HTML5 ja CSS3 ovat kaksi uutta standardia W3C eli World Wide Web Consortiumissa. Nämä kaksi ryhmää toimivat hyvin yhdessä. On jopa sanottu, että niiden käyttö on seuraava kehityssaskel jokapäiväisessä teknologiassa. Ne myös auttavat rakentamaan uusia innovoivampia ratkaisuja internet-sivustoille. Hogan (2010, 14) täsmentää HTML5:n uusien toiminnallisuuksien rakentuvan paremmin uusille alustoille verkkojärjestelmien kehityksessä. Näistä tärkeimpiä ovat merkityksellisemmät tagit ja käyttökokemuksen parantaminen animaatioilla sekä multimediatuella.

HTML-merkistökielen tueksi otettiin PHP-kieli, joka luo toivotun dynaamisuuden eli päivitettävyyden ja käytettävyyden. PHP-koodia tullaan implementoimaan eli sisällyttämään internet-sivuston etu- ja erityisaukioloaika-sivulle. Näin kohdeyrityksellä olisi mahdollista lähettää uusia tiedotuksia helposti ilman sivujen erillistä muokkausta.

CSS-merkistö tulee lopussa huolehtimaan sivuston ulkoasullisesta toteutuksesta. Tarkemmin käytetään CSS:n versiota 3 (CSS3). CSS3:n muokausmahdollisuudet esimerkiksi fonttien eli kirjainmahdollisuuksien kanssa on varteenotettava vaihtoehto erityisesti ulkoasun muokausmahdollisuuksien kanssa. CSS3 ja HTML5 ovatkin hyvä vaihtoehto, kun halutaan päästä lähelle Flash-ohjelmoinnin dynamiikkaa ja muokattavuutta.

3.4 Sisällönhallintajärjestelmä ja sen toiminta

Sisällönhallintajärjestelmän (eng. CMS; Content Management System) ydin on sen suunnittelussa. Verens (2010, 7) kuvailee sisällönhallintaa kuin tietokoneen osia. Emolevy on tietokoneen tärkein komponentti. Ilman sitä prosessori, muistit tai muut osat eivät toimisi. Sisällönhallinnan ydin on runko mikä yhdistää tietokannan, selaimen näyttämän sivuston ja kaikki muut toiminnot yhteen.

Sisällönhallinta -ja sisällönhallintajärjestelmättömiä sivustoja Verens (2010, 8) vertailee seuraavasti; sisällönhallinnattomassa sivustossa pyydetään selaimelta tiettyä osoitetta, jolloin se lähettää pyynnön palvelimelle. Palvelin toimittaa siten tiedon yhdestä tietystä sivusta. Sisällönhallinnan omaavassa sivussa, sivun arkkitehtuuri on erilainen. Sivun voi koostua monista eri elementeistä, kuten uutisista ja muista tiedotteista, jotka sisällönhallinta kokoaa yhteen.

Verens (2010, 8) tarkentaa CMS:n toimintaa tietokannan avulla. Tavanomaisesti kaikki tiedot sisällytetään HTML-tiedostoon, josta se jaetaan palvelimen kautta yhtenä pakettina. CMS:n kautta tuleva tieto haetaan tietokannan kautta ja prosessoidaan matkalla valittujen toimintojen mukaan luettavaksi tiedoksi loppukäyttäjälle näkyvällä sivustolla.

Verens (2010, 8) kertoo CMS:n yksityiskohtaisen toimintaperiaatteen seuraavasti. Palvelin lähettää pyynnön sisällönhallinnalle, josta sisällönhallinta rikkoo pyynnön omiksi osikseen. Sivun haetaan seuraavaksi tietokannan kautta. Jos sivu kuitenkin käyttää liitännäisiä eli erillistä ohjelmaa, jota selain ei automaattisesti osaa avata, suorittaa se ne tässä vaiheessa. Lopulta kaikki kerätty tieto kootaan yhteen ja internet-selain muuntaa haetun kokonaisuuden yhdeksi eheäksi HTML-dokumentiksi.

Sisällönhallinnan suunnittelu aloitettiin luomalla ensin MySQL-relaatiotietokannan. Relatiotietokanta tulisi pitämään sisällään kolme päätaulua eli osa-alueita, joiden sisälle tallennettaisiin jokaisen eri osa-alueen taulujen kannalta pääkäyttäjän lisäämät tiedot.

Toteutuksen kannalta parhaimmat osa-alueet olivat sisäänkirjautuminen, tiedotukset ja erillisaukioloajat. Niihin sisällytetyt kentät suunniteltaisiin mahdollisimman suorituskäytännöiksi eli vältettäisiin turhien kenttien lisäämistä ja näin suorituskäytännön menettämistä. Kirjautumistaulu tulee sisältämään kolme asiakenttää: henkilön nimen, käyttäjätunnuksen ja salasanan. Salasanat tullaan salaamaan MD5-koodauksella, joka tapahtuisi PHP:n avulla.

Tiedotukset-taulu tulisi sisältämään aikatieköt, jotka se hakee palvelimelta kun tiedotuksen lähetys tapahtuu. Aikatiekötön lisäksi taulu sisältäisi lähetettävän käyttäjän tunnuksen, jonka PHP-lähetää kirjautumisen yhteydessä. Näin voidaan seurata kuka on lisännyt tiedotuksen.

Viimeisinä ovat itse asiakentät, jotka ovat tiedotuksen otsikko ja sisältö. Erillisaukioloajat ovat viimeinen taulu, joka tulee sisältämään vain yhden tekstikentän, johon lisätään halutut aukioloajat.

Edellisten kenttien lisäksi on lisättävä vielä teknisen toimivuuden vuoksi jokaiselle osa-alueelle lisäkenttä tunnus, ID. ID on ainutlaatuinen kenttä. Kenttään lisätään vielä on autoincrement merkintä eli automaattisesti lisääntyvä uniikki numerointi. Näin lisätty sisältö ei voi mennä sekaisin ja sitä on helpompi hakea. (Tizag, 2008).

Sisäänkirjautuminen tulee toimimaan salatulla osoitteella, joka avaa luodun sisällönhallinnan. Pääsyvaatimusten mukaan avautuu valikko, josta halutut asiat voidaan tehdä. Sisällönhallinta koostuu tiedotusten lisäämisestä ja erillisaukioloaikojen muokkaamisesta.

Tiedotusten lisäys tapahtuu pääkäyttäjän toimesta valitsemalla uuden tiedotuksen lisääminen. Tämän jälkeen avautuu lomake, joka pyytää käyttäjää lisäämään tiedotuksen otsikon ja halutun viestin. Sivustolla on lista jo lähetetyistä tiedotuksista, joiden ominaisuuksiin kuuluu, muokkaus ja poisto. Erillisaukioloaikojen muokkaussivu sisältää lisätyt tiedot, joita voidaan myös muokata tai poistaa tarpeen tullen.

4 Toteutus

Toteutusalue rakentui aikaisemmin mainituista merkintäkielistä ja ohjelmista. Toteutusympäristössä tärkeimmät dynaamisen internet-sivun osat olivat PHP:n ja MySQL-relaatiotietokannan yhteistyö. Loppurakenne tehtiin käyttäen suurimmaksi osaksi HTML5-kieltä ja CSS3-merkintöjä. Lopputulos viimeisteltiin kuvankäsittelyn avulla tuotetulla ulkoasukokonaisuudella. Työssä auttoi suunnattomasti Notepad++ ohjelmisto, joka teki koodin kirjoittamisesta ja muokkaamisesta helpompaa ja myös yksinkertaisti prosessia.

4.1 MySQL

Kehitys aloitettiin luomalla MySQL-relaatiotietokanta, johon lisättiin kolme taulua; sisäänkirjautuminen, tiedotukset ja erillisaukioloajat. Taulujen rakenteessa otettiin huomioon toiminnalliset näkökulmat, kuten sisällön merkitseminen riippumattomaksi muusta lisäystä sisällöstä id:n, eli tunnisteen ja autoincrementin avulla. Näin ollen voitiin lisätä uusia tiedotuksia, käyttäjiä ja muokata valmiita erillisaukioloaikoja ilman niiden sekaantumista järjestelmässä.

MySQL-tietokannassa käytettiin InnoDB-moottoria, joka on MySQL-tietokannassa toimiva varastointimoottori. (Stackoverflow, 2010). Se valittiin sen vakauden ja riskittömyyden

vuoksi. Jos palvelin esimerkiksi kaatuu kesken lähetyksen, ei tietokannalle käy kuinkaan, vaan InnoDB automaattisesti viimeistelee aikaisemmat tapahtumat ennen kaatumista. (MySQL 2013). Myös relaatioiden tuottaminen on vaivattomampaa taulujen jako-ominaisuuden ansiosta.

Tarvittavia relaatioita olivat muunmuassa käyttäjätietojen tallentuminen lisättyihin tiedotteisiin ja erillisaukioloaikojen muokkaukseen. Näin ollen pääkäyttäjä voi seurata muiden käyttäjien tekemiä muutoksia halutessaan.

4.2 PHP

Jotta verkkosivu pystytään yhdistämään tietokantaan, luotiin sille oma php-tiedosto. Tiedosto sisälsi mysqli_connect toiminnon, joka yhdisti tietokantaan annetun käyttäjätunnuksen ja salasanan avulla. Lisäksi syötettiin tieto mitä tietokantaa käytetään. Yhdistämistiedosto sisälsi myös mysqli_error -tarkistuksen virheiden varalta.

Tietokantaan yhdistäviä metodeja on useita. Yleisin käytetty tapa on mysqli_connect-funktio. Mysqli_connectin ongelmana on kuitenkin sen heikko tietoturva hyökkäystilanteessa. Mysqli_connect lähettää kaiken datan tavallisessa tekstimuodossa, jolloin osaava hyökkääjä pystyy verkon kautta saamaan kaiken tarvittavan tiedon haltuunsa. (Rook, 2010). Käytettäväksi toiminnoksi valittiin MySQLi-funktio, jossa i tarkoittaa parannettua (eng. Improved).

Tietokannan luonnin jälkeen aloitettiin erillisten php-tiedostojen luominen. PHP-tiedostot ohjaavat sivun dynaamista kulkua tietokannan kautta. PHP-lomakkeiden pääasiallinen toiminta perustuu niihin syötettyihin tietoihin, jotka se sitten lähettää SQL-yhteyden kautta tietokantaan. Se hakee pyydettyä sisäänkirjautuneen käyttäjän käyttäjätiedot ja käyttäjän antaman sisällön ja lähettää ne lomakkeen mukana muuttujien kautta tietokannan haltuun. Prosessissa salasana salattiin tietokantaa varten MD5-metodilla.

Lomakkeita luotiin kolme kappaletta, joista ensimmäinen oli salattu uuden käyttäjän lisäyslomake. Koodin avulla voitiin määrittää, mitkä rekisteröitymisehdot sallittaisiin. Työssä luotiin ehtoja, kuten salasanan pituus, joka ei voi olla alle viisi kirjainta. Lisäksi tieto ei lähde eteenpäin, jos käyttäjä on antanut salasanaakseen oman käyttäjätunnuksensa. Kun kaikki tiedot ovat oikein, lisätään uusi käyttäjä tietokantaan. Tietoturvan vuoksi pääkäyttäjän on vielä erikseen aktivoitava uusi käyttäjä manuaalisesti tietokannan kautta.

Toiseksi luotiin lomake tiedotusten lisäämiseen. Käyttäjälle ilmestyy kirjautumisen jälkeen paneeli, jonka kautta voidaan lisätä uusi tiedotus. Uuden tiedotuksen lisäämisessä käyttäjän

tarvitsee vain lisätä tiedotuksen otsikko ja sisältö. PHP lisää automaattisesti käyttäjän nimen tietokantaan ja julkaisee lähetysajankohdan otsikon ja sisällön julkiselle internet-sivun pääsivulle. Tiedotuksia on mahdollista muokata kolmannella linkillä, joka löytyy navigoinnista.

Viimeinen lomake on erillisaukioloaikojen lisääminen. Lomaketta käytettäessä se sisällyttää automaattisesti viimeiset annetut tiedot, joita voidaan muokata tai poistaa. Käytettävyyden parantamiseksi muokkaustoiminto toimii vain yhden taulun varassa, joka voidaan pahimmassa tapauksessa vain tyhjentää.

4.3 JQuery

JQuery:llä luotiin paremmin toimiva tiedotteiden poistamiseen tarkoitettu toiminto. Tiedotteen muokkauksen viereen lisättiin poista-linkki, jota painamalla ilmestyy vahvistusikkuna. Ikkunassa kysytään halutaanko tiedostus poistaa vai perutaanko poistoyritys. Alkuperäisesti poistotoiminto toimi PHP:n avulla, mutta se korvattiin JQueryllä paremman käytettävyyden takaamiseksi.

Jqueryn avulla parannettiin sivuston toimivuutta tiedotuksen poiston jälkeen. Sivun täydellinen uudelleenlataaminen (eng. Refresh) on yleisin tapa päivittää sivusto poistotoiminnon jälkeen. Se kuitenkin rasittaa pitkällä tähtäimellä verkkosivua ylläpitävää palvelinta. Turhan tiedon kierrättämisen sijaan valittiin parempi tapa, joka päivittää Ajax-toiminnon avulla vain pyydetyn alueen sivustolta.

4.4 Ulkoasu

Ulkoasun määrittäminen tapahtui luomalla yrityksen toivomalla värimailmalla raikas ja piristävä ulkoasu. Mallina käytettiin aikaisemman sivustoversion värimailmaa ja osaa kuvista. Kohdeyrityksellä oli lukuisia eri tuotekuvia, jotka haluttiin sisällyttää ulkoasuun. Sivuston ajatuksena jatkui halu erottua nykyajan massatuotettujen sivujen rinnalla. Tyylillä haettiin yksinkertaista, mutta samalla mielenkiintoa ja ulkonäöllisesti värikästä kokonaisuutta. Tämä hoidettiin luomalla yksinkertainen sivupohja tuotekuvia käyttämällä.

4.5 Kuvankäsittely

Kuvankäsittelyssä käytettiin Adoben Photoshop CS6 -ohjelmistoa. Ulkoasu aloitettiin luomalla tyhjä 900x600 pikselin levyinen alue näytöllä. Suurempia resoluutioita varten sivun ulkoasua paisutettiin taustakuvan avulla, joka ei kuitenkaan häiritse mobiilikäyttäjiä skaalautuvuutensa

takia. Mittasuhteeseen päädyttiin matalaresoluutioisten näyttöjen takia, joita on esimerkiksi älypuhelimissa. Näin kaikki tieto näkyy myös niissä.

Lukuisien eri kirjasintyyppien joukosta päädyttiin käyttämään Jellyka Cake-nimistä kirjaisinta sivuston otsikoinniksi (eng. banner). Sivustojen ollessa ovat kaupalliset, pyydettiin tekijältä lupa kirjasintyyppin käyttämiseen ja tarjottiin lahjoitus kiitokseksi. Linkeissä käytettiin internetin standardikirjaisimia.

Ulkoasun värimaailma koostuu valkoisesta ja pastellinvihreän sekoituksesta. Otsikointi ”Herkkukonditoria-Leipä” muokattiin tekijän oman kokeilun mukaan kullanhohtavaksi pienin mustin reunasimien. Taustaksi luotiin koko sivun vaakasuorasti peittävä vihertävä borderi eli raja.

4.6 HTML ja CSS

Internet-sivuston rakenne aloitettiin luomalla emodivi eli sivuston rakenne elementti, joka sisältää yhden tai useamman elementin sisäänsä. (Web teacher, 2005). Emodivin (eng. Wrapper) sisälle asetettiin pienemmät class-div, jolla tarkoitetaan useasti käytettävää sivuston elementtirakennetta (Coyier, 2008). Näihin diveihin lukeutui banneri (suom. Otsake), linkit ja sisältö. Emodiville annettiin CSS:n avulla oma parametri position:relational, jonka avulla sivuston rakenne keskittää itsensä näytölle. Näin ollen kaikki emodivin sisällä olevat rakenteet liikkuvat sen mukana.

Sivuston rakenteen luomisen jälkeen diveihin sisällytettiin PHP:n require-komennon avulla sisältö. Emodivin sisällä toimii neljä diviä, joista jokainen hakee oman yksilöidyn sivun itselleen. Yhdessä nämä sivut luovat kokonaisuuden jokaisen yksilöidyn divin avulla. Lisäksi PHP-require komennolla haetut sivut sisältävät omat divi-rakenteensa, jotka rikastavat pääsivun toimintaa.

Yksilöityihin sivuihin lisättiin sisältö utf-8-koodattuna tekstinä. Sivuston ensimmäinen class-div on linkit. Linkejä varten luotiin omat kuvat, joihin sisällytettiin sivun navigaatio. Sivuston otsakekuva määritettiin CSS:n avulla toimimaan divin taustakuvana sille määrätyillä leveys ja korkeus -ehdoilla.

Sisällön kanssa käytetään jo valmista sisällönhallintasivun pohjaa, joka lisää tallennetun lisäyksen näkyväksi yrityksen pääsivuilla. Pääsivulle ei luonnollisesti lisätä muokkaus tai lisäysmahdollisuuksia, vaan haetaan tietokantaan lisätty tiedote valmiiseen sivupohjaan liitettynä. Yrityksen verkkosivulla näkyy viisi tuoreinta julkaisua. Vanhemmat julkaisut saadaan näkyville kohdistetun linkin kautta, joka hakee uudelle sivulle kaikki tiedotukset.

Rakenteen ollessa kunnossa, testattiin se ajanmukaisilla selaimilla. Selaimiin lukeutui Google Chrome, Mozilla Firefox, Internet Explorer 6-8 ja Safarin selain. Rakennevaiheessa on helpointa korjata tietyt viat, jotka voivat pilata ulkoasun toimivuuden loppuvaiheessa. Sivuston ollessa suunniteltu kunnolla ei määrittämisvirheitä näkynyt, jolloin voitiin jatkaa ulkoasun sisällyttämiseen sivustolle.

Sivuston ulkoasu toteutettiin valkoisen, vihreän ja keltaisen ympärillä. Näitä määrittämiä hoitaa erillinen CSS-tiedosto. CSS:n avulla luotiin erikseen sivun taustakuva background-no-repeat:n keinoin. Näin saatiin koko sivun mittainen vihertävä tausta sivulle. Taustan määrittäykset estävät sen liikkumisen sivun mukana, jolloin sivuston ulkoasullinen ilme ei mene rikki. Lisäksi tausta on suunniteltu olemaan tarpeeksi suuri, jotta se ei rikkoudu edes suurempia resoluutioita käytettäessä.

Thumbnailit eli esikatselukuvat haluttiin järjestää persoonallisemmin sivulle. Tätä varten käytettiin CSS3:n rotate-tekniikkaa. Rotate-tekniikan avulla voidaan säätää kuvan asetuskulmaa. Jokaisella selaimella on oma tapansa vastaanottaa tämänkaltaisia käskyjä. Transform käskyt asetettiin css-tiedoston sisälle tukemaan nykypäivän selaimia.

Tarkemmin CSS3-standardia käytettäessä sivulle suunniteltiin CSS3-lightbox eli animointiasetusta apuna käyttäen yksinkertainen valokuvien avausanimaatio. Yrityksen toiveiden mukaan sivustolle sisällytettiin ulkoasuun sopivia tuotekuvia valokuvateemaisesti. Ohjelmointi tapahtui luomalla luokkien avulla linkit kuviin niin, että ne CSS3-tiedoston avulla avautuvat animaation kanssa suuremmiksi. Sulkeminen tapahtuu ohessa olevan sulje-napin avulla. CSS-koodiin lisättiin vielä gradient eli erisävyinen tausta kuvan selauksen ajaksi.

CSS määrittämiä käytettiin aikaisempien asioiden lisäksi myös eri otsakkeiden luonnissa, jotta sivustolle saatiin ilmeen lisäksi luonnetta. Sisältötekstin väriksi valittiin musta, sillä se toimi parhaimmin vihertävän taustan kanssa. Lisäksi linkit luotiin vaihtamaan väriä rusahtavasta tummempaan sävyyn.

Lopullinen linkistön luonti sivulle määräytyi sisällyttämällä php-koodi indexin eli pääsivun rakenteeseen. Kyseinen php-määrittäminen kertoo sivustolle minkä sisällön se avaa yksittäiseen sisältöikkunaan sitä pyydettyäessä eikä sivu turhaan päivitty aina uudestaan. Tämänkaltaisen toimenpide säästää sekä palvelinta että käyttäjän verkkokaistaa. Toimenpide on suositeltavaa varsinkin mobiilikäyttäjiä silmällä pitäen.

5 Yhteenveto

Työssä oppitiin, että nykypäivän dynaamisesti toimivat verkkosivut on parasta rakentaa käyttäen useita eri ohjelmointikieliä yhdessä. Valitut HTML(5), CSS(3), PHP ja MySQL-relaatiotietokanta sopivat yhteen erittäin hyvin. Kehitettäessä näillä menetelmillä täytyy kuitenkin muistaa kehittämisjärjestys. Helpoin tapa aloittaa uuden sivuston rakentaminen on suunnitella ensin tietokanta, jonka ympärillä koko verkkosivusto toimii. Lisäksi on suositeltavaa suunnitella se alusta niin, että sivua on varaa laajentaa lisäominaisuuksia varten. Toimivan PHP-koodin testaus sekä koonti on seuraava askel. Kun sivuston tekniikka on saatu toimimaan siirrytään ulkoasun luomiseen kuvankäsittelyohjelmalla. Viimeiseksi askeleeksi saadaan sivuston rakenteen ja ulkoasumäärittysten yhteinen toiminta HTML:n ja CSS:n avulla. Tekijä piti erittäin tärkeänä edellämainittujen asioiden muistamista oman työnsä lopputuloksen kannalta.

Ennen lopputuotteen päästämistä julkiseen levitykseen, testattiin sitä useilla eri mobiililaitteilla. Sivusto suoritui sekä sisällönhallinnan toiminnallisuudella kuin myös kuluttajille suunnatulla yrityksen verkkosivustolla kaikkien mobiililaitteiden selaimilla toimivaksi kokonaisuudeksi. Sivuston skaalautuvuutta ja toiminallisuutta mitaroitiin tarkalleenottaen nykypäivän Meego, iOS6 ja Android -käyttöjärjestelmillä. Testissä toimi sekä tabletit että älypuhelimet. Luonnollisesti sivustoa testattiin myös tietokoneilla.

Työn aikana kohdeyritystä informoitiin joka askeleen valmistuttua. Lisäksi työn ollessa vasta teoriana pöydällä, voitiin kehittämis ehdotuksia ottaa vastaan. Näin ollen saatiin tilaajana toimivalle kohdeyritykselle sen haluama dynaamisesti toimiva ja mobiilivalmis verkkosivusto sisällönhallinnalla.

Tekijä uskoo sivuston olevan valmis nykypäivän käyttöä ja tarpeita varten. Sivuston toimiessa pohjaltaan HTML5-version avulla on se helposti valmis tuleville päivityksille sekä parannuksille. Myös tietokantaa luotaessa jätettiin tilaa mahdollisia uudistuksia ja parannuksia silmällä pitäen. Esimerkkinä käyttäjien seuraaminen. Sivuston ulkoasussa otettiin huomioon käyttäjäystävällisyys sekä selkeys. Kohdeyritys ilmaisi myös olevansa tyytyväinen prosessin läpiviemiseen sekä sivuston ulkoasun toimivuuteen brändinsä kanssa.

Toteutus alustan suunnittelu ja harkinta auttoi suunnattomasti lopputulokseen päätyemisessä. Lisäksi työn loppuvaiheessa löydettiin uusi parempi tapa päivittää sisällönhallintaa JQueryllä, joka vähensi palvelimen kuormitusta. Valitut lähteet vahvistivat oikean kehitystavan valinnan, jolloin työn loppukehitys sekä toimivan tuotteen toimittaminen kohdeyritykselle oli mieluista ja opettava kokemus.

Lähteet

Kirjalähteet

Grannel, C. Sumner, V. & Synodinos, D. 2012. The Essential Guide to HTML5 and CSS3 Web Design. Friendsoft, New York.

Hogan, B. 2010. HTML5 and CSS3 Develop with Tomorrows Standards Today. The Pragmatic Bookshelf. Dallas, Texas.

Yank, K. 2012. PHP and MySQL: Novice to Ninja. Sitepoint. Collingwood.

Thomson, L. & Welling, L. 2009. PHP and MySQL Web Development fourth edition. Pearson Education, Inc. New York.

Brampton, M. 2008. PHP5 CMS Framework Development. Packt Publishing Ltd. Birmingham.

Verens, K. 2010. CMS Design Using PHP and JQuery. Packt Publishing Ltd. Birmingham.

Snyder, C. Myer, T. & Southwell, M. 2010. Pro PHP Security. Apress. New York.

Cabral, K. & Murphy, K. 2009. MySQL Administrator's Bible. Wiley Publishing. Indianapolis.

Sähköiset lähteet

Introduction to HTML. 2013. W3Schools. Viitattu 14.3.2013

http://www.w3schools.com/html/html_intro.asp

Flash Professional CS6 FAQ. 2013. Adobe Flash. Viitattu 15.3.2013

<http://www.adobe.com/products/flash/faq.html>

Md5_file. 2013. PHP.net. Viitattu 15.3.2013

<http://www.php.net/manual/en/function.md5-file.php>

Steve Jobs shares his 'thoughts on Flash'. 2010. Techspot.com. Viitattu 17.3.2013

<http://www.techspot.com/news/38773-steve-jobs-shares-his-thoughts-on-flash.html>

HTML <div> Tag. 2013. W3Schools. Viitattu 17.3.2013

http://www.w3schools.com/tags/tag_div.asp

Mysql tables. 2008. Tizag. Viitattu 18.3.2013

<http://www.tizag.com/mysqlTutorial/mysqltables.php>

InnoDB as the default MySQL Storage Engine. 2013. MySQL. Viitattu 18.3.2013

<http://dev.mysql.com/doc/refman/5.5/en/innodb-default-se.html>

Introduction to JQuery - a Powerful Javascript Library. 2013. Oracle. Viitattu 25.3.2013

<http://www.oracle.com/technetwork/java/intro-jquery-166845.html>

JQuery: Advantages and Disadvantages. 2010. Jscripters. Viitattu 25.3.2013

<http://www.jscripters.com/jquery-disadvantages-and-advantages/>

Ajax load page and reload/add after success. 2012. Stackoverflow. Viitattu 3.4.2013

<http://stackoverflow.com/questions/13384989/ajax-load-page-and-reload-add-after-success>

Php mysql_connect security. 2010. Rook. Viitattu 3.4.2013

<http://stackoverflow.com/questions/3888065/php-mysql-connect-security>

CSS3 explained. 2012. CSS3explained. Viitattu 7.5.2013

<http://www.css3explained.com/>

CSS3 Lightbox. 2011. Mary Lou. Viitattu 9.5.2013

<http://tympanus.net/codrops/2011/12/26/css3-lightbox/>

What is programming. 2013. David Bolton. Viitattu 11.5.2013

<http://cplus.about.com/od/introductiontoprogramming/p/programmers.htm>

What is MySQL. 2013. Freeservers. Viitattu 11.5.2013

<http://www.freeservers.com/WebHosting101/WhatIsMySQL.html>

What is plugin-container. 2013. Mozilla Firefox support. Viitattu 11.5.2013

<http://support.mozilla.org/en-US/kb/What%20is%20plugin-container>

What is InnoDB and MyISAM in MySQL. 2010. Stackoverflow. Viitattu 11.5.2013

<http://stackoverflow.com/questions/3818759/what-is-innodb-and-mysam-in-mysql>

What is a wrapper div. 2005. Web teacher. Viitattu 11.5.2013

<http://www.webteacher.ws/2005/02/01/494/>

The difference between id and class. 2008. Chris Coyier. Viitattu 11.5.2013

<http://css-tricks.com/the-difference-between-id-and-class/>

Liitteet

Liite 1: Sisällönhallinnan etusivu	27
Liite 2: Tiedotteiden, lisäys, muokkaus ja poisto	28
Liite 3: Uuden tiedotteen lisäys	29
Liite 4: Tiedotteen muokkaus	30
Liite 5: Erillisaukioloajat esikatselu	31
Liite 6: Erillisaukioloaikojen muokkaus	32
Liite 7 Tiedotteen poisto	33
Liite 8: Herkkukonditoria-leipa.fi etusivu	34
Liite 9: Internet-sivuston kuvan esikatselu	35

Liite 1: Sisällönhallinnan etusivu

You are logged in as testeri

[Logout](#)

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Viimeisimmät tiedotukset

[kaikki tiedotukset](#)

Päivitetty kotisivusto - 02.04.13 - 16:24 -

Tervetuloa uudistuneelle **Herkkukonditoria-leipä.fi** sivustolle!

Sivusto tukee nyt kaikkia nykypäivän mobiililaitteita, joten sivustolle pääsy on helpompaa kuin koskaan ennen. Lisäksi sivulle on lisätty tiedotukset, joiden avulla ilmoitusten jättäminen on entistä helpompaa.

Sivusto on toteutettu opinnäytetyönä, jonne myös mediatiedote tullaan julkaisemaan. Nauttikaa siis uudesta nykypäivän käyttökokemuksesta!

[muokkaa](#)

Liite 2: Tiedotteiden, lisäys, muokkaus ja poisto

You are logged in as testeri

[Logout](#)

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Tiedotusten lisäys, muokkaus ja poisto

[Lisää uusi tiedotus](#)

Päivitetty kotisivusto - 02.04.13 - 16:24 - testeri

Tervetuloa uudistuneelle **Herkkukonditoria-leipä.fi** sivustolle!

Sivusto tukee nyt kaikkia nykypäivän mobiililaitteita, joten sivustolle pääsy on helpompaa kuin koskaan ennen. Lisäksi sivulle on lisätty tiedotukset, joiden avulla ilmoitusten jättäminen on entistä helpompaa.

Sivusto on toteutettu opinnäytetyönä, jonne myös mediatiedote tullaan julkaisemaan. Nauttikaa siis uudesta nykypäivän käyttökokemuksesta!

[muokkaa](#) - [poista](#)

Liite 3: Uuden tiedotteen lisäys

You are logged in as testeri

Logout

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Tiedotteen lisäys

aihe:

sisältö:

Tallenna

Peruuta

Liite 4: Tiedotteen muokkaus

You are logged in as testeri

Logout

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Tiedotteen muokkaus

aihe:

Päivitetty kotisivusto

sisalto:

```
<p>Tervetuloa uudistuneelle <b>Herkkukonditoria-  
leipä.fi</b> sivustolle!</p>  
  
<p>Sivusto tukee nyt kaikkia nykypäivän  
mobiililaitteita, joten sivustolle pääsy on  
helpompaa kuin koskaan ennen. Lisäksi sivulle on  
lisätty tiedotukset, joiden avulla ilmoitusten  
jättäminen on entistä helpompaa.</p>  
  
<p>Sivusto on toteutettu opinnäytetyönä, jonne myös  
mediatiedote tullaan julkaisemaan. Nauttikaa siis  
uudesta nykypäivän käyttökokemuksesta!</p>
```

Tallenna

Peruuta

Liite 5: Erillisaukioloajat esikatselu

You are logged in as testeri

[Logout](#)

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Erillisaukioloajat

Tässä muokataan aikoja

Aukioloaika 1

Aukioloaika 2

Aukioloaika 3

Aukioloaika 4

Aukioloaika 5

[muokkaa](#)

Liite 6: Erillisaukioloaikojen muokkaus

You are logged in as testeri

[Logout](#)

[Etusivu](#) - [Tiedotteiden lisäys, muokkaus ja poisto](#) - [Erilliaukioloajat](#)

Erillisaukioloaikojen muokkaus

Tässä muokataan aikoja

<p> Aukioloaika 1 </p>

<p> Aukioloaika 2 </p>

<p> Aukioloaika 3 </p>

<p> Aukioloaika 4 </p>

<p> Aukioloaika 5 </p>

[Tallenna](#)

[Peruuta](#)

Liite 8: Herkkukonditoria-leipa.fi etusivu



Liite 9: Internet-sivuston kuvan esikatselu

